1. The example in Section 8.6.4 from the white book (which is also in one of the slides) shows that covariant subtyping over object types is unsound in the general case (i.e. without restricting method update). Construct an example showing that contravariant subtyping over object types is also unsound in the general case (i.e. without restricting method selection).

2. (This exercise is a follow-up of Exercise 3 from Problem Set 8) Consider the object term \( o \triangleq [\ell = \varsigma(s : A)s] \).

   (a) Is there a type \( A \) in \( \text{Ob}_1^{<} \) with variance annotations such that the term \( o.\ell.\ell \) is typable?

   (b) Can we generalize this result for any number of selections? I.e., for a the term \( o.\ell \ldots \ell \) where the number of \( \ell \)'s is fixed but unbounded.

3. The third rule in Table 11.1 (Recursive Algorithm Unify) applies the substitution \( \{ t \mapsto \tau \} \) (or using the notation in the book \( [\tau/t] \)) to the set \( E \) before the recursive call to Unify whenever \( t \not\equiv \tau \). Suppose we re-write the algorithm so that this substitution is not applied to \( E \) at all (but only returned as part of the final substitution due to the use of \( \circ \)). Show that, in this case, algorithm Unify may return a substitution that does not unify \( E \).

4. Exercise 11.2.8 parts (a), (b) and (c).

5. Exercise 11.2.14 parts (a), (b) and (c).